

© EPODOC / EPO

PN - JP2003103845 A 20030409
PD - 2003-04-09
PR - JP20010297440 20010927
OPD - 2001-09-27
TI - PRINTING CONTROLLING UNIT, CONTROLLING METHOD THEREFOR, AND PROGRAM
IN - YAMAMOTO TAKESHI
PA - CANON KK
IC - B41J5/30 ; B41J29/42 ; G06F3/12

© WPI / DERWENT

TI - Printing controller has display unit which displays printing and non-printing proportion within predetermined range that corresponds to number of printed pages and plain pages
PR - JP20010297440 20010927
PN - JP2003103845 A 20030409 DW200333 B41J5/30 011pp
PA - (CANO) CANON KK
IC - B41J5/30 ;B41J29/42 ;G06F3/12
AB - JP2003103845 NOVELTY - A display unit displays the advance situation of the printing process. When the printing is not completed, the display unit graphically displays the printing and non-printing proportion within predetermined range. The range corresponds to number of printed pages and predetermined number of plain pages. When the printing is completed, the range represents the remaining number of plain pages.
- DETAILED DESCRIPTION - INDEPENDENT CLAIMS are also included for the following:
- (1) printing control method; and
- (2) printing control program.
- USE - For controlling printing process of printer.
- ADVANTAGE - The completion of the paper roll is informed in advance to the user.
- DESCRIPTION OF DRAWING(S) - The figure shows the block diagram of a printer driver. (Drawing includes non-English language text).
- (Dwg.2/10)
OPD - 2001-09-27
AN - 2003-348207 [33]

© PAJ / JPO

PN - JP2003103845 A 20030409
PD - 2003-04-09
AP - JP20010297440 20010927
IN - YAMAMOTO TAKESHI
PA - CANON INC

THIS PAGE BLANK (USPTO)

- TI - PRINTING CONTROLLING UNIT, CONTROLLING METHOD THEREFOR, AND PROGRAM
- AB - PROBLEM TO BE SOLVED: To provide a printing controlling unit, capable of displaying the progress state of despooling naturally in a form close to the reality even in the case of controlling despooling of the printing job so as to start without awaiting finish of spooling, a controlling method therefor, and a program.
 - SOLUTION: Whether or not spooling is finished during despooling is judged. While spooling is not finished, the ratio of despooling and undespooling is displayed graphically according to the ratio of the number of despoiled pages and a predetermined number of undespoiled pages. After finish of spooling, the area representing the predetermined number of pages in the predetermined area is displayed as the area representing the remaining undespoiled pages.
- I - B41J5/30 ;B41J29/42 ;G06F3/12

THIS PAGE BLANK (USPTO)

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開2003-103845

(P2003-103845A)

(43)公開日 平成15年4月9日(2003.4.9)

(51)Int.Cl. ⁷	識別記号	F I	テームト*(参考)
B 4 1 J 5/30		B 4 1 J 5/30	Z 2 C 0 6 1
29/42		29/42	F 2 C 0 8 7
G 0 6 F 3/12		G 0 6 F 3/12	C 5 B 0 2 1

審査請求 未請求 請求項の数6 O L (全 11 頁)

(21)出願番号 特願2001-297440(P2001-297440)

(22)出願日 平成13年9月27日(2001.9.27)

(71)出願人 000001007

キヤノン株式会社

東京都大田区下丸子3丁目30番2号

(72)発明者 山本 剛

東京都大田区下丸子3丁目30番2号 キヤ
ノン株式会社内

(74)代理人 100076428

弁理士 大塚 康徳 (外3名)

Fターム(参考) 2C061 AP01 AP03 AP04 AQ05 CQ27
CQ34

2C087 AA03 AA09 AB01 AB05 AC07

BA07 BA12 BD36 CB16

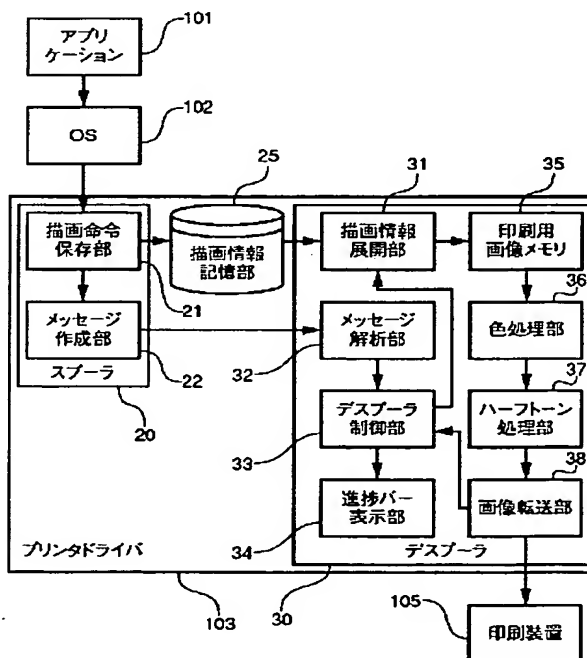
5B021 AA01 CC02 CC07 DD00 PP05

(54)【発明の名称】 印刷制御装置およびその制御方法ならびにプログラム

(57)【要約】

【課題】 スプールの完了を待たずにその印刷ジョブのデスプールの開始するように制御する場合にも、デスプールの進行状況を、より自然でかつ、実際に近い態様で表示することができる印刷制御装置およびその制御方法ならびにプログラムを提供すること。

【解決手段】 デスプール中にスプールが完了したか否かを判断し、スプールが完了していない間は、デスプールおよび未デスプールの割合を、デスプールされたページ数と未デスプールとしての所定ページ数との割合によってグラフィカル表示し、スプールが完了した後は、前記所定領域内の前記所定ページ数を示す領域を、未デスプールの残ページ数を示す領域と見立てて表示する。



【特許請求の範囲】

【請求項1】 印刷データをスプールするとともに、スプールされた印刷データをデスプールする印刷制御装置であって、

スプールが完了したか否かを判断する判断手段と、

デスプールの進行状況を表示する表示手段と、

を備え、

前記表示手段は、

スプールが完了していない間は、デスプールおよび未デスプールの割合を、デスプールされたページ数と未デスプールとしての所定ページ数との割合によって所定領域内にグラフィカル表示し、

スプールが完了した後は、前記所定領域内の前記所定ページ数を示す領域を、未デスプールの残ページ数を示す領域として設定し、表示することを特徴とする印刷制御装置。

【請求項2】 印刷データをスプールするとともに、スプールされた印刷データをデスプールする印刷制御装置の制御方法であって、

スプールが完了したか否かを判断する判断ステップと、

デスプールの進行状況を表示する表示ステップと、

を有し、

前記表示ステップは、

スプールが完了していない間は、デスプールおよび未デスプールの割合を、デスプールされたページ数と未デスプールとしての所定ページ数との割合によって所定領域内にグラフィカル表示し、

スプールが完了した後は、前記所定領域内の前記所定ページ数を示す領域を、未デスプールの残ページ数を示す領域として設定し、表示することを特徴とする制御方法。

【請求項3】 印刷データをスプールするとともに、スプールされた印刷データをデスプールする印刷制御装置を制御するためのプログラムであって、

スプールが完了したか否かを判断する判断ステップのプログラムコードと、

デスプールの進行状況を表示する表示ステップのプログラムコードと、

を有し、

前記表示ステップのプログラムコードは、

スプールが完了していない間は、デスプールおよび未デスプールの割合を、デスプールされたページ数と未デスプールとしての所定ページ数との割合によって所定領域内にグラフィカル表示し、

スプールが完了した後は、前記所定領域内の前記所定ページ数を示す領域を、未デスプールの残ページ数を示す領域として設定し、表示することを特徴とするプログラム。

【請求項4】 印刷データをスプールするとともに、スプールされた印刷データをデスプールする印刷制御装置

であって、

スプールが完了したか否かを判断する判断手段と、

デスプールの進行状況を表示する表示手段と、

を備え、

前記表示手段は、

スプールが完了していない間は、デスプールおよび未デスプールの割合を、デスプールされたページ数と未デスプールとしての所定ページ数との割合によって所定領域内にグラフィカル表示し、

10 スプールが完了した後は、前記所定領域内の前記所定データサイズを示す領域を、未デスプールの残データサイズを示す領域として設定し、表示することを特徴とする印刷制御装置。

【請求項5】 印刷データをスプールするとともに、スプールされた印刷データをデスプールする印刷制御装置の制御方法であって、

スプールが完了したか否かを判断する判断ステップと、

デスプールの進行状況を表示する表示ステップと、

を有し、

20 前記表示ステップは、

スプールが完了していない間は、デスプールおよび未デスプールの割合を、デスプールされたページ数と未デスプールとしての所定ページ数との割合によって所定領域内にグラフィカル表示し、

スプールが完了した後は、前記所定領域内の前記所定データサイズを示す領域を、未デスプールの残データサイズを示す領域として設定し、表示することを特徴とする制御方法。

30 【請求項6】 印刷データをスプールするとともに、スプールされた印刷データをデスプールする印刷制御装置を制御するためのプログラムであって、

スプールが完了したか否かを判断する判断ステップのプログラムコードと、

デスプールの進行状況を表示する表示ステップのプログラムコードと、

を有し、

前記表示ステップのプログラムコードは、

スプールが完了していない間は、デスプールおよび未デスプールの割合を、デスプールされたページ数と未デスプールとしての所定ページ数との割合によって所定領域内にグラフィカル表示し、

40 スプールが完了した後は、前記所定領域内の前記所定データサイズを示す領域を、未デスプールの残データサイズを示す領域として設定し、表示することを特徴とするプログラム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、印刷処理の進行状況を表示する技術に関する。

【0002】

【従来の技術】プリンタドライバは、プリンタで印刷を行うために以下のような処理を行っている。

【0003】まず、ユーザがアプリケーションで印刷実行を指示すると、アプリケーションから複数の入力描画命令がOSを介してプリンタドライバに渡される。プリンタドライバはこの入力描画命令を解析して描画に必要な情報から印刷用スプールデータを作成してスプールファイルに保存する。全ての入力描画命令がスプールされると、実際に印刷を行うデスプール処理が開始される。デスプール処理ではこのスプールファイルに保存した印刷用スプールデータを取り出して印刷画像を作成し、この印刷画像から印刷コマンドを作成してプリンタに送ること

で実際の印刷を行っている。

【0004】プリンタドライバは画面に表示されるウィンドウを持っていて、ユーザはこのウィンドウのGUI (Graphical User Interface) を介して印刷処理の監視や制御を行うことが可能となっている。プリンタドライバが表示するウィンドウはスプールされた全ての印刷ジョブに関する情報を表示するスプーラウィンドウと、デスプール中のジョブに関する情報を表示するデスプーラウィンドウがある。

【0005】スプーラウィンドウでは、スプールされている印刷ジョブを確認することができ、その印刷ジョブに対して一時停止やキャンセル、処理順序の変更などの制御が可能である。デスプーラウィンドウでは現在デスプール中の印刷ジョブを確認することができ、印刷処理の一時停止やキャンセルが可能となっている。

【0006】また、デスプーラウィンドウには現在デスプールしているジョブの印刷処理の進行度合いの確認のために、進捗バーが表示される。印刷ジョブの処理の進行度合いは一般に、進捗バーの領域全体に対するバーの塗りつぶされた割合で示される。つまり、印刷が開始されたときバーは全く描画されないが、処理が進むにつれて徐々に伸びていき、印刷データの処理が半分終了すれば進捗バーは全体の1/2まで進み、ジョブが終了するときには領域全体が塗りつぶされることになる。

【0007】進捗バーを描画する長さを決める方法は種々のものが提案されている。1つはスプールデータのデータサイズを基準としたもので、スプールファイルのサイズを全体として印刷処理の終了したデータサイズの割合で描画する長さを決める方法がある。この他、印刷ジョブによって印刷される総ページ数を全体として印刷処理の終了したページ数の割合で描画する長さを決める方法もある。

【0008】

【発明が解決しようとする課題】しかしながら、従来行なわれていた方法では、以下のような問題点がある。

【0009】従来は、スプール処理が全て終了してからデスプール処理が開始されていたので、デスプール処理の開始時にはスプールファイルのサイズや印刷ジョブの

総ページ数を知ることができた。しかし、近年では印刷時間を短縮するために、スプール処理が全て終了するのを待つのではなく、1ページ目のスプール処理が終了したら1ページ目のデスプール処理を開始することが行われている。この場合、1ページ目のデスプール処理終了時に全ページのスプール処理が終了しているとは限らず、スプールファイルサイズや全ページ数を知ることができず、進捗バーを正確に描画することはできない。

【0010】例えば、現在のスプールファイルサイズに対して、処理の終了したスプールデータサイズの割合を使用して進捗バーを描画すると、デスプール処理自体は進行しているのに、スプール処理が進行してスプールファイルサイズが大きくなると進捗バーは短くなってしまい、進捗バーの動作としては適切ではない。これはスプールの終了したページ数に対する処理の終了したページ数の割合を使用して進捗バーを描画した場合も同様である。

【0011】本発明は、上記の問題点を解決するためになされたもので、スプールの完了を待たずにその印刷ジョブのデスプールを開始するように制御する場合にも、デスプールの進行状況を、より自然でかつ、実際に近い態様で表示することができる印刷制御装置およびその制御方法ならびにプログラムを提供することを目的とする。

【0012】

【課題を解決するための手段】本発明によれば、例えば以下の構成を備える印刷制御装置が提供される。すなわち、印刷データをスプールするとともに、スプールされた印刷データをデスプールする印刷制御装置であって、スプールが完了したか否かを判断する判断手段と、デスプールの進行状況を表示する表示手段と、を備え、前記表示手段は、スプールが完了していない間は、デスプールおよび未デスプールの割合を、デスプールされたページ数と未デスプールとしての所定ページ数との割合によって所定領域内にグラフィカル表示し、スプールが完了した後は、前記所定領域内の前記所定ページ数を示す領域を、未デスプールの残ページ数を示す領域として設定し、表示することを特徴とする。

【0013】

【発明の実施の形態】以下、図面を参照して本発明の好適な実施形態について詳細に説明する。

【0014】図1は、実施形態に係る画像形成システムの構成例を示すブロック図である。本システムは同図に示すように、ホストコンピュータ（以下、単に「ホスト」ともいう）100に、例えば、インクジェットプリンタなどのプリンタ105、および、ディスプレイ106が接続された構成である。

【0015】ホストコンピュータ100は、ホスト全体の動作を制御するCPU 108、主記憶装置として機能するRAM 109、ブートプログラム等を記憶しているROM 110、そ

して、各種プログラム等を格納するハードディスク(HDD) 107を備える。

【0016】また、101は、ワードプロセッサ、表計算、インターネットブラウザなどのアプリケーションソフトウェア、102はオペレーティングシステム(OS)、103はプリンタ105を制御するプリンタドライバ、104はディスプレイ106を制御するディスプレイドライバである。それぞれ、HDD107にインストールされているものであるが、電源投入後にRAM109にロードされ、現在CPU108によって実行中であることを示している。

【0017】アプリケーションソフトウェア101によって発行される出力画像を示す描画命令は、OS 102を介してディスプレイドライバ104に入力される。また、印刷を行う場合、その描画命令はOS 102を介してプリンタドライバ103に入力される。プリンタドライバ103はその描画命令を処理して印刷データを作成しプリンタ105に印刷させるソフトウェアであり、ディスプレイドライバ104は前記描画命令を処理して表示画像を作成してディスプレイ106に表示させるためのソフトウェアである。

【0018】OS 102はアプリケーションソフトウェア101から印刷出力要求を受けると、プリンタ105に対応するプリンタドライバ103に描画命令を渡す。プリンタドライバ103はOS 102から渡される印刷出力要求および描画命令群を処理して、プリンタ105が印刷処理可能な印刷データを作成し、その印刷データをプリンタ105に送る。プリンタ105がラスタプリンタである場合、プリンタドライバ103は、描画命令を、順次、例えばRGBそれぞれ8ビットの深さをもつ印刷用画像メモリに展開する。そして、全ての描画命令をラスタライズした後、印刷用画像メモリの内容をプリンタ105が印刷可能なコマンドに変換してプリンタ105に送る。なお、印刷用画像メモリは例えばRAM 109に割り当てられる。

【0019】図2は、プリンタドライバ103の詳細な構成例を示している。プリンタドライバ103は、図示の如く、スプーラ20、描画情報記憶部25、および、デスプーラ30を備えている。

【0020】スプーラ20において、21はアプリケーションソフトウェア101からの描画命令を保存する描画命令保存部、22はデスプーラ30に送るメッセージを作成するメッセージ作成部である。

【0021】描画情報記憶部25は描画命令保存部21からの描画情報を保存する。

【0022】デスプーラ30において、31は描画情報記憶部25から描画情報を読み出して画像としてラスタライズする描画情報展開部、32はメッセージ作成部22によって作成されたメッセージを受け取って解析するメッセージ解析部、33はデスプーラ全体の制御を司るデスプーラ制御部、34はウィンドウ上に進捗バーを表示する進捗バー表示部、35は描画情報展開部31によってラスタライズされる印刷用画像を記憶する印刷用画像メモリ、36はラス

タライズ画像の色補正を行なう色処理部、37はパターンディザや誤差拡散などの手法によってハーフトーン処理を行なうハーフトーン処理部、38は画像データをプリンタコマンドに変換してプリンタ105に転送する画像転送部、である。

【0023】実施形態に係る画像形成システムの構成は概ね上記のとおりである。次に、スプーラ処理について説明する。

【0024】図3は、実施形態におけるプリンタドライバ103のスプーラ20における処理の一例を示すフローチャートである。なお、本明細書に添付する図面のフローチャートにおいて、記号「=」は、右辺の値を左辺に代入する意味で用いるものとする。ただし、判断ブロックにおいて用いられる記号「=」は、右辺と左辺とが等しいことを示す等号として扱うものとする。

【0025】まず、初期化处理として処理ページのカウンタ変数であるPageを1に設定し(ステップS101)、デスプーラ30を起動する(ステップS102)。

【0026】その後、カウンタ変数Pageで示されるページについて描画命令保存処理を行い(ステップS103)、メッセージ作成部22でデスプーラ30にページ保存終了メッセージを発行して、パラメタとしてPageの値をデスプーラ30に渡し(ステップS104)、Pageをカウンタアップする(ステップS105)。

【0027】上記ステップS103~105の処理を、全ページが終了するまで繰り返し(ステップS106)、全ページ終了したらメッセージ作成部22でデスプーラにジョブ保存終了メッセージを発行する(ステップS107)。

【0028】図4は、上記したステップS103における描画命令保存処理の詳細を示すフローチャートである。

【0029】アプリケーション101がユーザからの印刷指示に基づき描画命令を発行すると(ステップS201)、OS 102はプリンタドライバ103の能力に応じて描画命令を変換してプリンタドライバ103に渡し(ステップS202)、プリンタドライバ103は受け取った描画命令を描画命令保存部21で描画命令記憶部25に保存する(ステップS203)。上記ステップS201~203の処理を、該当ページの全描画命令が終了するまで繰り返す(ステップS204)ことで描画命令を保存する。

【0030】続いて、デスプーラ処理について説明する。

【0031】図5は、プリンタドライバ103のデスプーラ30における処理を示すフローチャートである。

【0032】まず、初期化处理として、デスプーラ処理中のページ番号を示す変数PageRを1に、スプーラ済みのページ数を示す変数PageSを0に、ジョブのスプーラ完了フラグであるEndFlagをFALSEに、進捗バーの長さであるPhを0に設定し(ステップS301)、メッセージの処理(詳細は後述)を行う(ステップS302)。

【0033】その後、デスプーラの進行状況を示す進捗

バーの表示処理(ステップS303)、描画命令再生処理を行い(ステップS304)、PageRをカウントアップした後(ステップS305)、メッセージの処理を行い(ステップS306)、次ページの保存が終了したかどうかをチェックし(ステップS307)、終了していれば上記ステップS303に戻って処理を繰り返す。次ページの保存が終了していなければ、全ページの処理が終了したかどうかをチェックし(ステップS308)、終了していなければステップS306に戻り、次ページの保存が終了するか、または、全ページが終了するまでステップS306のメッセージの処理を繰り返す。全ページ終了した時点でデスブル処理は終了する。

【0034】図6は、上記ステップS304における描画命令再生処理の詳細を示すフローチャートである。

【0035】まず、描画情報展開部31で描画情報記憶部25から描画情報を取得して印刷用画像メモリ35にラスタライズする(ステップS401)ことを、該当ページ的全描画情報に対して行う(ステップS402)。

【0036】全描画情報のラスタライズが終了したら、印刷用画像メモリ35に格納された画像に対して色処理部36で色処理を、ハーフトーン処理部37でハーフトーン処理を行い(ステップS403)、画像転送部38でプリンタのコマンドに変換してプリンタ105へ送る(ステップS404)。そして、描画情報記憶部25に保存された該当ページの描画情報を削除する(ステップS405)。

【0037】以上の処理によって1ページの印刷処理を実現する。

【0038】図7は、上記したステップS302におけるメッセージの処理の詳細を示すフローチャートである。

【0039】メッセージ解析部32はスプーラ20のメッセージ作成部22からのメッセージを受け取るまで待ち(ステップS501)、受け取ったメッセージを解析し(ステップS502)、ページ保存終了メッセージであれば、メッセージのパラメタとして渡された保存終了ページをPageSに設定する(ステップS503)。ジョブ保存終了メッセージであれば、EndFlagにTUREを設定する(ステップS504)。

【0040】次に、上記のステップS303における進捗バーの表示処理について詳しく説明する。

【0041】進捗バーは、所定の表示領域内において、デスブルの進行状況に応じてその長さを変えて表示されるものである。例えば、デスブルがまだ全く行われていなければ、進捗バーは表示されない。デスブルが全体の半分まで行われた時には表示領域の半分が進捗バーで塗りつぶされる(残りの領域が未デスブルであることが分かる)。そして、デスブルが完了した時に表示領域全体が塗りつぶされる、という態様が自然で好ましい。

*

$$Ph = (H - Remain) + Remain \times (PageR - PageD) / (PageS - PageD)$$

(2)

*【0042】しかしながら、上述したとおり、スプールの完了を待たずにデスブルを開始する場合には、デスブル開始時点では最終的なスプールデータサイズが不明であるのだから、実際にはその実現は簡単ではない。

【0043】そこで、本発明者は、デスブル中において、スプールが完了していない間と、スプールが完了した後とで進捗バーの更新基準を変更して、より自然な表示態様に近づけることに成功した。

【0044】図8は、上記したステップS303における進捗バーの表示処理の詳細を示すフローチャートである。

【0045】まず、EndFlagがTRUEかFALSEかをチェックすることで現在の印刷ジョブのスプールが完了したかどうかを判断する(ステップS601)。ここで、スプールが完了していなければ、総ページ数は現在デスブル処理を行っているページ番号PageRより例えば5ページ多いとして進捗バーの長さPhを求める(ステップS602)。進捗バー表示領域の長さをHとすると、本ステップでの計算式は次式で表すことができる。

$$Ph = H \times PageR / (PageR + 5) \quad (1)$$

【0047】ところで、この場合、進捗バー表示領域の残りの領域の長さH-Phが、未デスブル分に対応する。そうすると、この進捗バーによる表示態様は、デスブルおよび未デスブルの割合を所定領域(進捗バー表示領域)にグラフィカルに表示したものであるともいえる。

【0048】そして、スプールが未完である間は、デスブルと未デスブルとの割合を、デスブルされたページ数(PageR)と未デスブルとしての所定ページ数(5ページ)との割合によって表示する。

【0049】次に、ステップS601でスプールが完了と判断された場合には、まず、スプール完了後最初の進捗バー表示かどうかを判断し(ステップS603)、最初の表示であれば、スプール完了前に描画されていない進捗バーの長さを示す変数RemainにH-Phを設定する(ステップS604)。その後、Phが0かどうかを判断し(ステップS605)、0でなければ、スプール完了前にデスブルが終了したページ数を示す変数PageDにPageRを設定し(ステップS606)、0であれば、PageDに0を設定する(ステップS607)。

【0050】そして、スプール完了後の進捗バーの長さを求める(ステップS608)。スプール完了後の進捗バーは、スプール完了前に描画していなかった領域を残りのページ数で等分した長さずつ描画する。この場合の計算式は以下のように表せる。

【0051】

【0052】つまり、スプールが完了した後は、進捗バー表示領域の残りの領域（すなわち、所定ページ数を示す領域）を、未デスプールの残ページ数を示す領域として見立て（設定し）、以降進捗バーの描画更新を行っていく。

【0053】こうして求めた長さPhの進捗バーを描画する（ステップS609）ことで進捗バーの表示処理を行う。スプール中の進捗バーの長さを求める計算式(1)によれば、デスプール処理が進むに従って徐々に大きくなり、また進捗バー領域の長さHよりも大きくなることはない。1ページあたりの進捗バーの描画量は総ページ数が少ないときには長く、多いときには短くなるので、自然かつ合理的な表示となる。

【0054】また、スプール終了後の進捗バー表示はスプール終了時に描画していない領域を残りページで等分するため、処理が1ページ進むごとに新たに塗りつぶされる進捗バー領域は一定となり、かつ、全ページ終了したときにちょうど進捗バー領域全体が塗りつぶされることになるので、自然かつ合理的な表示となる。

【0055】図5のステップS307における、次ページの保存が終了したかどうかのチェックは、次ページのページ番号を示しているPageRとスプール済みページ番号を示しているPageSとを比較して判断する。PageR ≤ PageSであれば次ページの保存が終了していると判断される。

【0056】図5のステップS308における、全ページの処理が終了したかどうかのチェックは、以下のようにして行う。

【0057】スプールが終了している場合のスプール済みページ番号は、該当ジョブの全ページ数を示している。したがって、以下の条件を満たすとき、全ページのデスプール処理が終了したと判断できる。

【0058】
EndFlag = TRUE かつ PageR > PageS (3)

【0059】図10は、実施形態における進捗バーの表示例を示す図である。

【0060】同図において、左斜線部はスプール中に描画する進捗バーを表し、右斜線部はスプール終了後に更新描画される進捗バーを表し、何も塗られていない部分は進捗バーが描画されていない部分を表す。なお、実際の表示では右斜線部および左斜線部は区別なく同一のパターンおよび色で塗りつぶされるようにしてもよい。また、進捗バーの下に書かれている数字は描画された進捗バーの長さ（進捗バーの描画されていない部分の長さ）の比率を表す。この数字は説明のためのものであり、実際には表示しなくてもよい。以下、詳細について説明する。

【0061】同図中、(A)はスプール中に1ページ目の印刷処理を行っている場合の進捗バー表示を示している。この場合、式(1)により、塗りつぶされる長さは進

捗バーの領域の1/6、つまり塗りつぶされる領域の長さ（塗りつぶされていない領域の長さ）の比は1:5となる。

【0062】同様に、(B)はスプール中に2ページ目の印刷処理を行っている場合の進捗バー表示であり、式(1)により、塗りつぶされる領域の長さ（塗りつぶされていない領域の長さ）の比は2:5となる。

【0063】(C)はスプール中に3ページ目の印刷処理を行っている場合の進捗バー表示で、式(1)により、塗りつぶされる領域の長さ（塗りつぶされていない領域の長さ）の比は3:5となる。

【0064】ここで、本印刷ジョブの全ページ数は6ページであるところ、3ページ目の印刷処理中にスプールが終了したと仮定する。(D)はこの場合の進捗バーの表示を示している。変数Remainに進捗バーの領域の塗りつぶされていない領域の長さを保存する。また、現在は3ページ目について印刷処理を行っているので、変数PageNに3が設定される。そして、式(2)によって進捗バーの長さが計算されるが、この場合は特に表示の変化は起こらない。

【0065】(E)はスプール終了後に4ページ目の印刷処理を行っている場合の進捗バー表示を示している。この場合、式(2)により、スプール終了後に塗りつぶされる長さはスプール終了時に塗りつぶされていなかった領域の1/3、つまりスプール終了時に塗りつぶされていなかった領域に対して、塗りつぶされる領域の長さ（塗りつぶされていない領域の長さ）の比は1:2となる。

【0066】同様に、(F)はスプール終了後に5ページ目の印刷処理を行っている場合の進捗バー表示であり、式(2)により、スプール終了後に塗りつぶされる長さはスプール終了時に塗りつぶされていなかった領域の2/3、つまりスプール終了時に塗りつぶされていなかった領域に対して、塗りつぶされる領域の長さ（塗りつぶされていない領域の長さ）の比は2:1となる。

【0067】(G)は最終ページの6ページ目の印刷処理を行っている場合の進捗バー表示であり、式(2)により、塗りつぶされる長さはスプール終了時に塗りつぶされていなかった領域の3/3、つまり進捗バー領域の全体が塗りつぶされることになる。

【0068】以上説明した実施形態によれば、スプール中の進捗バー表示はジョブの総ページ数をデスプール処理しているページ番号より例えば5ページ多いとして計算し、スプール終了後の進捗バー表示はスプール終了前に描画していない領域を残りのページ数で等分して描画することにより、スプール中に印刷する場合においても自然かつ適切な進捗バー表示が可能になる。

【0069】なお、上述した実施形態では、描画処理が終了したページ数に連動して進捗バーを描画するようにしたが、ページ数ではなく、描画処理が終了したデータサイズに連動して進捗バーを描画するようにしてもよい。

【0070】図9は、上記したステップS303において、描画処理が終了したデータサイズに連動して進捗バーを描画するようにした、別の態様による処理を示すフローチャートである。

【0071】まず、EndFlagがTRUEかFALSEかをチェックすることで現在のジョブのスプールが終了しているかどうかを判別する(ステップS701)。ここで、スプールが終了していなければ、総スプールデータサイズは現在デスプール処理を行っているページまでの描画情報記憶部25に保存されたスプールデータサイズの合計SizeRより例えば1MBytes大きいとして進捗バーの長さPhを求める(ステップS702)。進捗バー表示領域の長さをHとすると、本ステップでの計算式は次式で表すことができる。

【0072】

$$Ph = H \times \text{SizeR} / (\text{SizeR} + 2^{20}) \quad (1)' \quad *$$

$$\text{Ph} = (H - \text{Remain}) + \text{Remain} \times (\text{SizeR} - \text{SizeD}) / (\text{SizeS} - \text{SizeD}) \quad (2)'$$

【0076】こうして求めた長さPhの進捗バーを描画する(ステップS709)ことで進捗バーの表示処理を行う。スプール中の進捗バーの長さを求める計算式(1)'によれば、デスプール処理が進むに従って徐々に大きくなり、また進捗バー領域の長さHよりも大きくなることはない。1ページあたりの進捗バーの描画量は総ページ数が少ないときには長く、多いときには短くなるので、自然かつ合理的な表示となる。

【0077】また、スプール終了後の進捗バー表示はスプール終了時に描画していない領域を残りページのスプールデータサイズで比例配分するため、処理が1ページ進むごとに新たに塗りつぶされる進捗バー領域は該当ページのデータサイズに比例し、かつ、全ページ終了したときにちょうど進捗バー領域全体が塗りつぶされることになるので、自然かつ合理的な表示となる。

【0078】なお、上述した実施形態では式(1)で5ページ(式(1)'を用いる場合にあっては1MBytes)を加算することとしたが、5ページ(1MBytes)という加算数は一例であって、別の加算数を設定してもよい。この加算数を大きくするとスプール中に進捗バーを塗りつぶす領域が狭くなるので、総ページ数(総データサイズ)が多いと予想されるようなシステムでは大きめの数字を選択するとよいであろう。

【0079】

【他の実施形態】以上、本発明の実施形態を詳述したが、本発明は、複数の機器(例えばホストコンピュータ、インタフェイス機器、リーダ、プリンタ等)から構成されるシステムに適用しても、1つの機器からなる装置(例えば、複写機、ファクシミリ装置等)に適用してもよい。

【0080】なお、本発明は、前述した実施形態の機能を実現するソフトウェアのプログラム(図3から7ま

*【0073】ステップS701において、スプールが終了していると判断された場合には、まず、スプール終了後最初の進捗バー表示かどうかを判断し(ステップS703)、最初の表示であれば、スプール終了前に描画されていない進捗バーの長さを示す変数RemainにH-Phを設定する(ステップS704)。その後、Phが0かどうかを判断し(ステップS705)、0でなければ、スプール終了前にデスプールが終了したデータサイズを示す変数SizeDにSizeRを設定し(ステップS706)、0であれば、SizeD<0を設定する(ステップS707)。

【0074】そして、スプール終了後の進捗バーの長さを求める(ステップS708)。スプール終了後の進捗バーは、スプール終了前に描画していなかった領域を残りのページのスプールデータサイズで比例配分した長さずつ描画する。この場合の計算式は以下のように表せる。

【0075】

で、ならびに図8または9に示すフローチャートに対応したプログラム)を、システムあるいは装置に直接あるいは遠隔から供給し、そのシステムあるいは装置のコンピュータがその供給されたプログラムを読み出して実行することによっても達成される場合を含む。

【0081】したがって、本発明の機能処理をコンピュータで実現するために、そのコンピュータにインストールされるプログラムコード自体も本発明を実現するものである。つまり、本発明の特許請求の範囲には、本発明の機能処理を実現するためのコンピュータプログラム自体も含まれる。

【0082】その場合、プログラムの機能を有していれば、オブジェクトコード、インタプリタにより実行されるプログラム、OSに供給するスクリプトデータ等、プログラムの形態を問わない。

【0083】プログラムを供給するための記憶媒体としては、例えば、フロッピー(登録商標)ディスク、光ディスク(CD-ROM、CD-R、CD-RW、DVD等)、光磁気ディスク、磁気テープ、メモリカード等がある。

【0084】その他、プログラムの供給方法としては、インターネットを介して本発明のプログラムをファイル転送によって取得する態様も含まれる。

【0085】また、本発明のプログラムを暗号化してCD-ROM等の記憶媒体に格納してユーザに配布し、所定の条件をクリアしたユーザに対し、インターネットを介して暗号化を解く鍵情報を取得させ、その鍵情報を使用することで暗号化されたプログラムを実行してコンピュータにインストールさせて実現することも可能である。

【0086】また、コンピュータが、読み出したプログラムを実行することによって、前述した実施形態の機能が実現される他、そのプログラムの指示に基づき、コンピュータ上で稼働しているOS等が実際の処理の一部ま

たは全部を行い、その処理によって前述した実施形態の機能が実現され得る。

【0087】さらに、記憶媒体から読み出されたプログラムが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書き込まれた後、そのプログラムの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるCPU等が実際の処理の一部または全部を行い、その処理によっても前述した実施形態の機能が実現される。

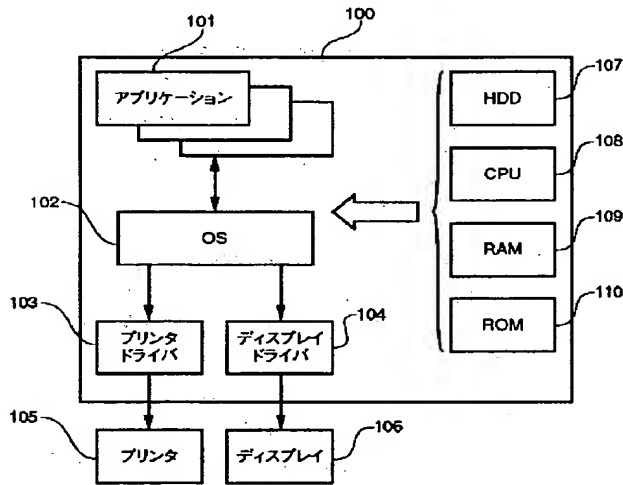
【0088】

【発明の効果】以上説明したように本発明によれば、スプールの完了を待たずにその印刷ジョブのデスプールの開始するように制御する場合にも、デスプールの進行状況を、より自然でかつ、実際に近い態様で表示することができる印刷制御装置およびその制御方法ならびにプログラムを提供することができる。

【図面の簡単な説明】

【図1】実施形態に係る画像形成システムの構成例を示すブロック図である。 *

【図1】



*【図2】実施形態におけるプリンタドライバの構成を示すブロック図である。

【図3】実施形態におけるスプールの処理を示すフローチャートである。

【図4】実施形態におけるスプールの描画命令保存処理の詳細を示すフローチャートである。

【図5】実施形態におけるデスプールの処理を示すフローチャートである。

10 【図6】実施形態におけるデスプールの描画命令再生処理の詳細を示すフローチャートである。

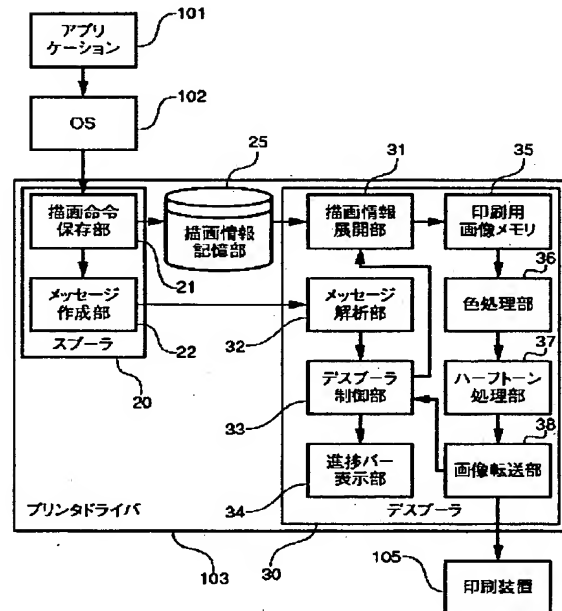
【図7】実施形態におけるデスプールのメッセージの処理の詳細を示すフローチャートである。

【図8】実施形態におけるデスプールの進捗バーの表示処理の詳細を示すフローチャートである。

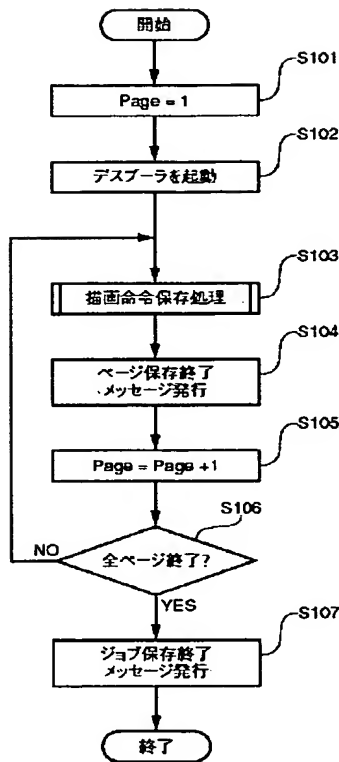
【図9】実施形態における、別の態様によるデスプールにおける進捗バーの表示処理を示すフローチャートである。

【図10】実施形態における進捗バーの表示例を示す図である。

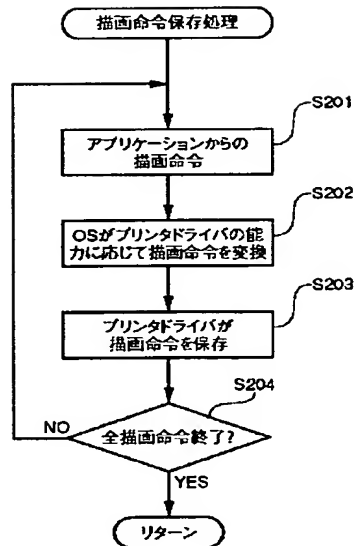
【図2】



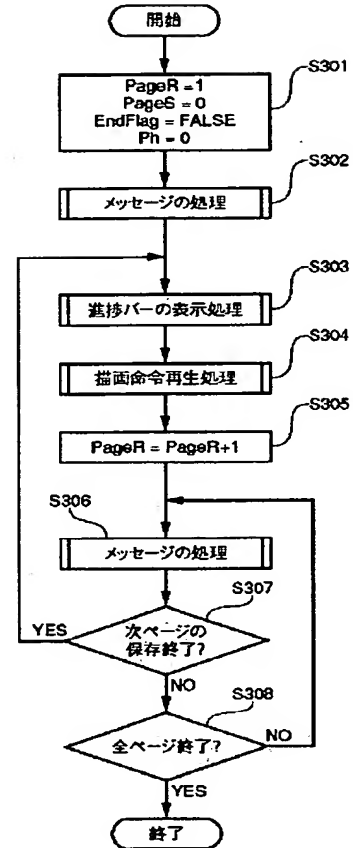
【図3】



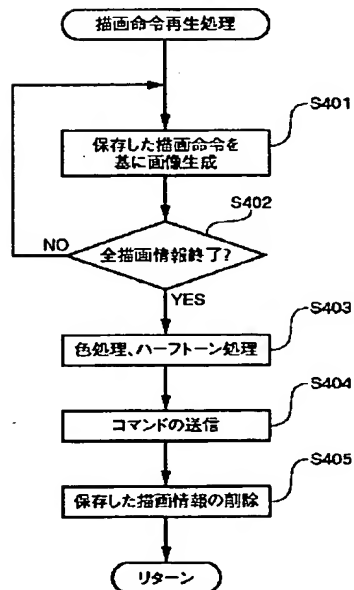
【図4】



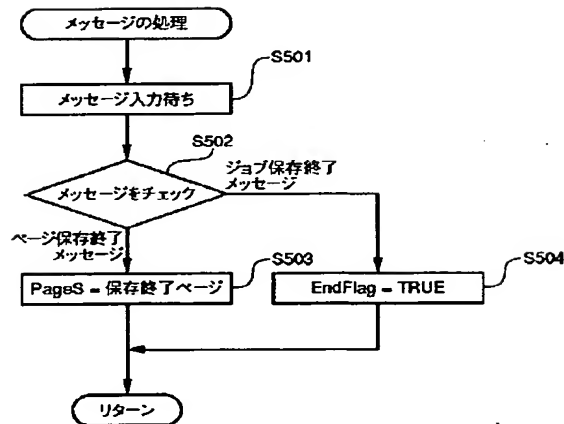
【図5】



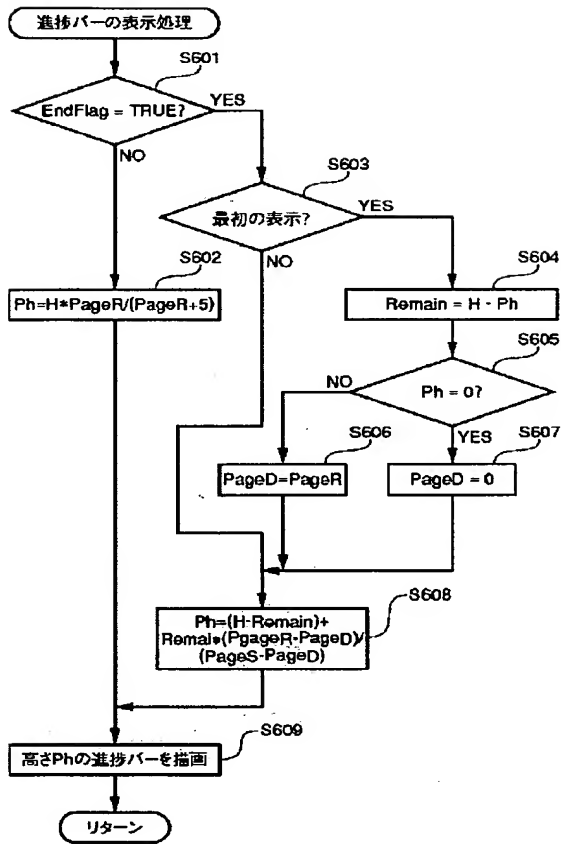
【図6】



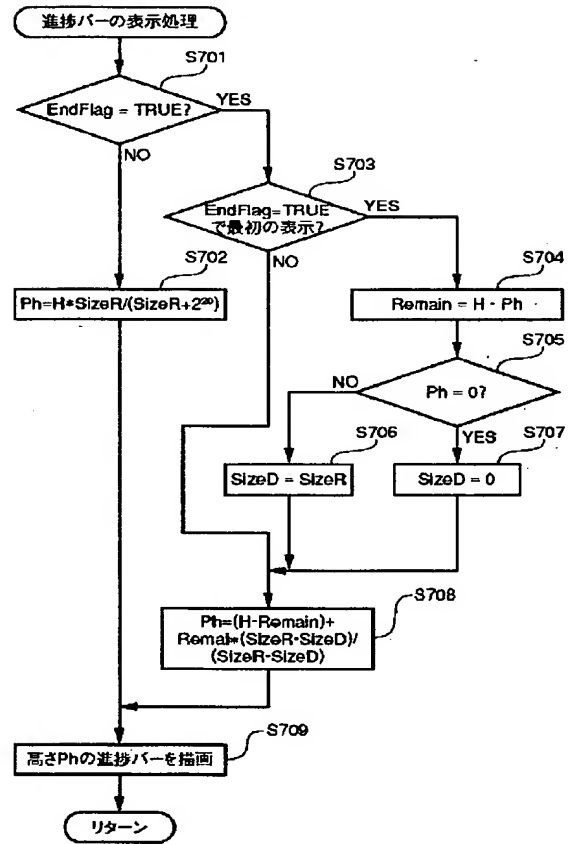
【図7】



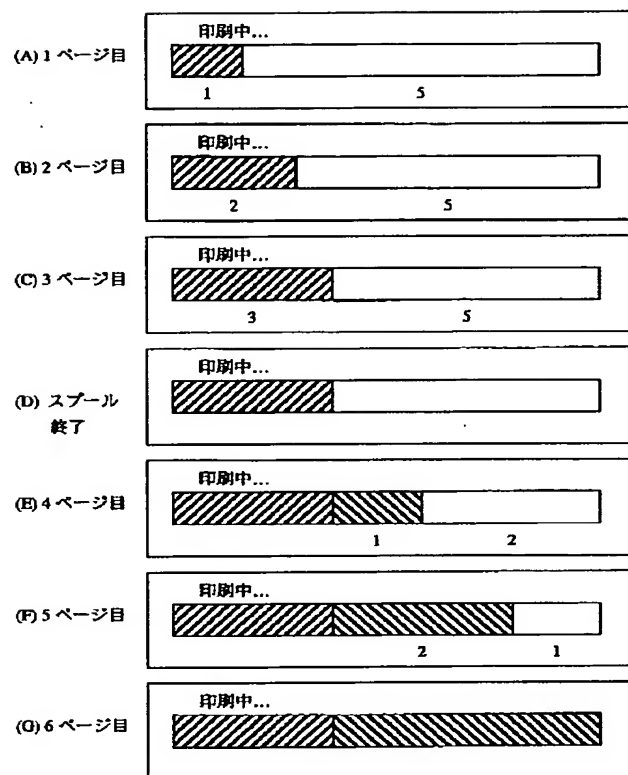
【図8】



【図9】



【図10】



THIS PAGE BLANK (USPTO)